**Таблица каскадов стилей. CSS3 транформирование.**

Введение в CSS трансформации

CSS трансформации используют для изменения вида, поворота и других преобразований элемента. Все эти настройки выставляются в обычных таблицах стилей, и когда вы открываете страницу, то уже видите готовый результат трансформации.

Функции трансформации

**Matrix -** задаёт матрицу преобразований.

**Rotate -** поворот элемента на заданный угол относительно точки трансформации, задаваемой свойством transform-origin.

transform: rotate(<угол>)

**Scale** - масштаб элемента по горизонтали и вертикали.

transform: scale(sx[, sy]);

Значение больше 1 увеличивает масштаб элемента, меньше 1 — уменьшает масштаб.

**ScaleX** - масштабирует элемент по горизонтали.

transform: scaleX(sx);

**ScaleY** - масштабирует элемент по вертикали.

transform: scaleY(sy);

**SkewX** - наклоняет элемент на заданный угол по вертикали.

transform: skewX(<угол>)

**SkewY** - наклоняет элемент на заданный угол по горизонтали.

transform: skewY(<угол>)

**Translate** - сдвигает элемент на заданное значение по горизонтали и вертикали.

transform: translate(tx[, ty])

**TranslateX** - сдвигает элемент по горизонтали на указанное значение. Положительное значение сдвигает вправо, отрицательное влево.

transform: translateX(tx)

**TranslateY** - сдвигает элемент по вертикали на указанное значение. Положительное значение сдвигает вниз, отрицательное вверх.

Примеры в файле transform.html

**Переходы CSS3**

Обратите внимание: данное свойство поддерживается в браузерах IE 10+, Chrome, Firefox и Opera. Для браузера Safari требуется добавить префикс -webkit.

С помощью нового CSS3 свойства **transition** Вы можете создавать эффекты перехода.

Для просмотра результата перехода будем использовать псевдокласс **:hover.**

Для создания переходов необходимо указать какое CSS свойство будет изменятся и скорость выполнения этих изменений в секундах, также нужно указать функцию смегчения. Для того, чтобы добавить эффект перехода к нескольким свойствам просто перечислите их названия через запятую.

Функции смягчения **transition-timing-function**

Плавность выполнения переходов контролируется с помощью функций смягчения. В CSS3 существуют несколько видов таких функций:

* linear
* ease (функция смягчения по умолчанию)
* ease-in
* ease-out
* ease-in-out
* cubic-bezier(x,x,x,x) (поведение функции контролируется переданными параметрами)

Свойство на которое распространяется трансформация -  **transition-property.**

* all — все
* width, color – только ширина и цвет текста(т.п. выборки)
* none — никто

Время, в течении которого происходит трансформация (указывается в секундах или в миллисекундах) - **transition-duration**.

Время, по истечении которого происходит трансформация(указывается в секундах или в миллисекундах) - **transition-delay**.

**Transition-timing-function** - указывает как должны изменяться промежуточные значения в кривой времени. Например, начинать развиваться медленно, а потом ускоряться. Трансформация может происходить и плавно на протяжении всего участка времени [cubic-bezier], и рывками в заданное количество шагов [steps].

Примеры в файле transition.html

**Градиенты в CSS3**

Линейные градиенты

**Обратите внимание**: градиенты поддерживается во всех современных браузерах, но требует добавления специального префикса. Для браузера IE10+ требуется префикс -ms, для Chrome и Safari префикс -webkit, для Opera префикс -o и для Firefox префикс -moz.

В CSS2.1 градиенты реализовывались в виде отдельных картинок вставляющихся как фоновые, в CSS3 имеются встроенные свойства для создания градиентов. Так как в CSS3 сам браузер отрисовывает градиенты необходимость дополнительных запросов градиентных картинок у сервера отпадает и это позволяет увеличить скорость загрузки страниц.

Линейные градиенты создаются с помощью CSS3 метода **linear-gradient**, который должен указываться в значение свойства **background**.

Для того, чтобы создать линейный градиент необходимо указать его направление (может задаваться с помощью ключевых слов или градусов) и цвета перехода.

Линейные градиенты могут иметь неограниченное количество цветов перехода.

Вы можете указывать координаты местоположения цветов с помощью % (0% подразумевает начало градиента, 100% конец).

Сферические градиенты

С помощью метода **radial-gradient** Вы можете создавать сферические градиенты.

Синтаксис определения сферических градиентов очень похож на синтаксис линейных, но требует также задания формы градиента (может быть сферической или эллипсоидной).

Повторяющиеся градиенты

Повторяющиеся градиенты задаются с помощью CSS3 методов **repeating-linear-gradient** (создает повторяющийся линейный градиент) и **repeating-radial-gradient** (создает повторяющийся сферический градиент).

Для того, чтобы создать повторяющийся градиент Вы должны указать направление градиента, а также цвета перехода и расстояние, которое они должны занимать.

Примеры в файле gradient.html

**CSS3 Анимация**

Обратите внимание: свойства анимации поддерживаются браузерами IE10+, Firefox и Opera. Для браузеров Chrome и Safari перед свойством требуется добавить префикс -webkit.

Для создания анимации в CSS3 используется свойство **@keyframes**.

Данное свойство представляет собой контейнер, в который должны помещаться различные свойства оформления.

Свойства анимации

|  |  |
| --- | --- |
| Свойство | Описание |
| @keyframes | Контейнер для определения анимации. |
| animation | Позволяет задать все значения для настройки выполнения анимации за одно определение. |
| animation-name | Позволяет указать имя анимации. |
| animation-duration | Позволяет задать скорость выполнения анимации в секундах (по умолчанию имеет значение 0). |
| animation-timing-function | Позволяет задать функцию смягчения отвечающую за плавность выполнения анимации (по умолчанию имеет значение ease). |
| animation-delay | Позволяет задать задержку перед началом выполнения анимации (по умолчанию имеет значение 0). |
| animation-iteration-count | Позволяет задать количество повторов анимации (по умолчанию имеет значение 1). |
| animation-direction | При значении alternate в нечетные разы (1,3,5 ...) анимация будет проигрываться в нормальном, а в четные (2,4,6 ...) в обратном порядке. По умолчанию данное свойство имеет значение normal, при данном значении анимация всегда проигрывается в нормальном порядке. |

Примеры в файле animation.html

**Разбиение текста на столбцы**

Обратите внимание: данные свойства поддерживаются в браузерах IE10+ и Opera. Для браузеров Chrome и Safari перед свойством требуется добавить префикс -webkit, а для Firefox префикс -moz.

С помощью CSS3 свойства **column-count** Вы можете указать количество столбцов, на которые необходимо разбить текст выбранного элемента.

С помощью CSS3 свойство **column-gap** Вы можете установить величину отступа между столбцами текста.

С помощью свойства **column-rule** Вы можете задать ширину, цвет и стиль оформления пространства между столбцами.

CSS3 свойство **column-width** позволяет указывать ширину столбцов текста

Пример в файле colum.html